AMENDMENTS TO THE CLAIMS

Cancel Claims 1, 3, 18 and 20 without prejudice. Please accept amended Claims 2 and 19 as follows:

1. (Cancelled)

2. (Currently Amended) The interleaver of claim ~~1~~ 21, wherein the number of seed variables is less than a size of a data block.

3. (Cancelled)

4. (Previously Presented) A turbo decoding system comprising:

a block interleaver;

an address queue for storing a generated interleaved address that is equal or smaller than a size of a data block;

an SISO decoder performing recursive decoding and calculating log likelihood ratio; and

an LLR memory connected to the SISO decoder and storing the log likelihood ratio, wherein the block interleaver comprises a preprocessor for preparing seed variables and an address generator for generating an interleaved address on the fly using the seed variables, wherein the SISO decoder accesses the data block and the log likelihood ratio in a sequential order and in an interleaved order alternately by the generated interleaved address,

wherein the generated interleaved address is once stored in the address queue and reused as a write address for writing the log likelihood ratio outputted from the SISO decoder into the LLR memory.

5. (Cancelled)

6. (Original) The turbo decoding system of claim 4, wherein the length of the address queue is equal to the SISO latency.

7. (Previously Presented) The turbo decoding system of claim 4, wherein the seed variables include a base column vector, an increment column vector, a cumulative column vector, and a modulo base, the number of elements of all three column vectors is equal to the number of rows of the block interleaver, and the elements of the column vectors are arranged by inter-row permutation, and wherein the cumulative column vector is updated by adding the increment vector to the old cumulative vector, then the interleaved addresses for one column are generated by adding the base vector and a vector that is calculated from the cumulative vector, and wherein ~~if~~ elements of the updated cumulative column vector larger than the modulo base are reduced by the modulo base.

8. (Previously Presented) A turbo decoding system comprising:

a processor for generating interleaved addresses, wherein the processor is a single-instruction multiple-data (SIMD) processor;

an address queue for storing the interleaved addresses;

a buffer memory block including an LLR memory for storing log likelihood ratio and a plurality of memory blocks for storing soft inputs; and

an SISO decoder connected to the buffer memory block, the SISO decoder including an ACSA network for calculating a log likelihood ratio recursively from soft inputs and the log likelihood provided by the LLR memory and a plurality of memory blocks connected to the ACSA network.

9. (Previously Presented) The turbo decoding system of claim 8, wherein the processor prepares seed variables when an interleaver structure changes due to a change of a coding standard or bit rate, and generates the interleaved addresses column by column using the seed variables when the interleaved addresses are required.

10. (Previously Presented) The turbo decoding system of claim 9, wherein the seed variables include a base column vector, an increment column vector, a cumulative column vector, and a modulo base, the number of elements of all three column vectors is equal to the number of rows of the interleaver, and the elements of the column vectors are arranged by inter-row permutation, and wherein the cumulative column vector is updated by adding the increment vector to the old cumulative vector, then the interleaved addresses for one column are generated by adding the base vector and a vector that is calculated from the cumulative vector, and wherein ~~if~~ elements of the updated cumulative column vector are larger than the modulo base are reduced by the modulo base.

11. (Previously Presented) The turbo decoding system of claim 8, wherein the SISO decoder supports a Viterbi decoding mode, wherein in Viterbi decoding mode, the ACSA network performs Viterbi recursion, the LLR memory stores traceback information outputted by the

ACSA network, the processor performs a traceback from the traceback information read from the LLR memory, and one of the memory blocks of the SISO decoder stores a path metric outputted by the ACSA network.

12. (Original) The turbo decoding system of claim 10, wherein the processor uses STOLT, SUBGE, LOOP instructions for fast calculation of interleaved addresses.

13-14. (Cancelled)

15. (Previously Presented) The turbo decoding system of claim 8, wherein the SIMD processor includes five processing elements for parallel processing, and wherein one of five processing elements controls the other four processing elements, processes scalar operation, and fetches, decodes, and executes instructions including control and multi-cycle scalar instructions, and wherein the other four processing elements only execute SIMD instructions.

16. (Previously Presented) The turbo decoding system of claim 8, wherein the SIMD processor includes five processing elements, and wherein one of five processing elements controls the other four processing elements, processes scalar operation, and fetches, decodes, and executes instructions including control and multi-cycle scalar instructions, and wherein the other four processing elements only execute SIMD instructions.

17. (Original) The turbo decoding system of claim 9, wherein the generated interleaved address is reused as a write address for writing the log likelihood ratio outputted from the SISO decoder into the LLR memory.

18. (Cancelled)

19. (Currently Amended) The interleaving method of claim ~~18~~ 22, wherein the seed variables are prepared when an interleaver structure changes due to a change of the coding standard or bit rate, and the interleaved addresses are generated column by column using the seed variables when the interleaved addresses are required.

20. (Cancelled)

21. (Previously Presented) An interleaver for rearranging sequences of data blocks in a data processing system, the interleaver comprising:

a preprocessor for preparing seed variables that vary according to the interleaving method of each of a plurality of standards and bit rates; and

an address generator means for generating an interleaved address on the fly using the seed variables,

wherein the seed variables include a base column vector, an increment column vector, a cumulative column vector, and a modulo base, the number of elements of all three column vectors is equal to the number of rows of the interleaver, and the elements of the column vectors are arranged by inter-row permutation in advance at the preprocessing, and

wherein the cumulative column vector is updated by adding the increment vector to an old cumulative vector, then the interleaved addresses for one column are generated by adding the base vector and a vector that is calculated from the cumulative vector, and

wherein elements of the updated cumulative column vector larger than the modulo base are reduced by the modulo base.

22. (Previously Presented) An interleaving method for rearranging a data block in a data communication system, comprising:

preparing seed variables; and

generating interleaved addresses column by column using the seed variables,

wherein the seed variables include a base column vector, an increment column vector, a cumulative column vector, and a modulo base, the number of elements of all three column vectors is equal to the number of rows of the interleaver, and the elements of the column vectors are arranged by inter-row permutation, and

wherein the cumulative column vector is updated by adding the increment vector to the old cumulative vector, then the interleaved addresses for one column are generated by adding the base vector and a vector that is calculated from the cumulative vector, and

wherein elements of the updated cumulative column vector are larger than the modulo base are reduced by the modulo base.